

Article



Improved Heart Disease Prediction Using Particle Swarm Optimization Based Stacked Sparse Autoencoder

Ibomoiye Domor Mienye 🗈 and Yanxia Sun *

Department of Electrical and Electronic Engineering Science, University of Johannesburg, Johannesburg 2006, South Africa; ibomoiyem@uj.ac.za

* Correspondence: ysun@uj.ac.za

Abstract: Heart disease is the leading cause of death globally. The most common type of heart disease is coronary heart disease, which occurs when there is a build-up of plaque inside the arteries that supply blood to the heart, making blood circulation difficult. The prediction of heart disease is a challenge in clinical machine learning. Early detection of people at risk of the disease is vital in preventing its progression. This paper proposes a deep learning approach to achieve improved prediction of heart disease. An enhanced stacked sparse autoencoder network (SSAE) is developed to achieve efficient feature learning. The network consists of multiple sparse autoencoders and a softmax classifier. Additionally, in deep learning models, the algorithm's parameters need to be optimized appropriately to obtain efficient performance. Hence, we propose a particle swarm optimization (PSO) based technique to tune the parameters of the stacked sparse autoencoder. The optimization by the PSO improves the feature learning and classification performance of the SSAE. Meanwhile, the multilayer architecture of autoencoders usually leads to internal covariate shift, a problem that affects the generalization ability of the network; hence, batch normalization is introduced to prevent this problem. The experimental results show that the proposed method effectively predicts heart disease by obtaining a classification accuracy of 0.973 and 0.961 on the Framingham and Cleveland heart disease datasets, respectively, thereby outperforming other machine learning methods and similar studies.

Keywords: deep learning; heart disease; particle swarm optimization; softmax regression; stacked sparse autoencoder

1. Introduction

Cardiovascular disease (CVD) is a life-threatening condition. It is the leading cause of death globally, with 30% of all global deaths attributed to it, amounting to 17 million deaths globally [1]. Additionally, deaths related to CVDs are estimated to rise to 22 million in 2030 if urgent measures are not taken [2]. Statistics from the American Heart Association (AHA) show that about 50% of American adults suffer from cardiovascular diseases [3]. Meanwhile, it is difficult to detect heart disease due to numerous contributory risk factors, including high blood pressure, diabetes, high cholesterol, arrhythmia, etc. [4]. However, these risk factors can be reduced through appropriate lifestyle changes.

Common symptoms of heart disease include shortness of breath, swollen feet, and body weakness [5]. Early detection is difficult but can significantly improve patient survival rates. Therefore, enhanced detection through machine learning (ML) based predictive models has been recently supported by clinicians to minimize the death rate and enhance the clinical decision-making process. The use of machine learning in clinical decision making can aid clinicians to detect heart disease risk and provide necessary treatments and recommendations to manage the risk [2]. To achieve this, electronic health records have been used to train ML models and learn the hidden relationships in the data. A few publicly available heart disease datasets and numerous predictive models have been



Citation: Mienye, I.D.; Sun, Y. Improved Heart Disease Prediction Using Particle Swarm Optimization Based Stacked Sparse Autoencoder. *Electronics* 2021, *10*, 2347. https:// doi.org/10.3390/electronics10192347

Academic Editors: Juan M. Corchado and Rashid Mehmood

Received: 26 July 2021 Accepted: 25 August 2021 Published: 25 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). developed over the years [6–10]. Meanwhile, researchers and scientists still have difficulty obtaining high prediction performance and identifying the most relevant heart disease risk factors [11].

The growing prevalence and high heart disease mortality rate necessitate rapid and accurate diagnosis of the disease [12]. Meanwhile, several ML research studies have focused on developing data preprocessing and transformation techniques to improve classification performance [13–15]. A typical example of such a data transformation technique is feature learning [16–18]. Recent advances in deep neural networks (DNN) have provided ML researchers with an opportunity to improve performance in different predictive models. Furthermore, the effectiveness of DNN can be utilized in learning the hidden correlations between the various features (i.e., risk factors) in heart disease datasets that would lead to improved prediction performance.

A set of salient features can be obtained by applying feature learning techniques, and the resulting low-dimensional feature representation can simplify the classification task. Feature learning or representation learning ensures the model automatically discovers the required representation for effective feature detection and classification of the input data [19]. Recently, feature learning techniques have received much attention, especially autoencoders, which have achieved state-of-the-art performance in learning a good representation of data [20]. An autoencoder (AE) can efficiently process and extract hidden representations from heart disease data for proper classification.

The network learns a hidden correlation between the input features and reconstructs it at the output. Furthermore, a sparsity constraint can be imposed on the hidden units that enable the network to better represent the input [19,21,22], thereby allowing the supervised learning algorithm to perform better classification. Meanwhile, determining the optimal parameters to train the feature learning algorithm is essential in achieving excellent deep feature learning [23,24]. Therefore, this paper presents an efficient heart disease prediction approach using the particle swarm optimization technique to optimize the parameters of a stacked sparse autoencoder. Optimizing the parameters of the SSAE ensures the algorithm achieves efficient feature learning.

We also aim to prevent internal covariate shift [25], a problem associated with multilayer autoencoders such as the proposed SSAE, by introducing batch normalization to the network. Finally, the effectiveness of the proposed approach is demonstrated by comparing it with other well-known ML methods, including logistic regression (LR), k-nearest neighbor (KNN), decision tree, linear discriminant analysis (LDA), support vector machine (SVM), extreme gradient boosting (XGBoost), adaptive boosting (AdaBoost), random forest, softmax classifier, and some methods in recent literature.

The rest of the paper is organized as follows. Section 2 briefly reviews some related works. The dataset used in the experiment, the concepts incorporated in the proposed approach, together with the proposed framework, are described in Section 3. Section 4 presents and discuss the simulation results, and Section 5 concludes the paper.

2. Related Works

This section briefly presents some recently proposed related works, including heart disease prediction models and relevant autoencoder studies. Predicting heart disease using machine learning can effectively reduce the number of deaths attributed to the disease. Recently, researchers have employed ML to detect heart disease and identify the most relevant risk factors [26,27]. For example, Mohan et al. [4] proposed an enhanced heart disease prediction model to identify the essential features in the data that would improve the prediction accuracy. The authors employed a hybrid random forest that achieved a classification accuracy of 88.7%.

Haq et al. [28] employed a feature selection approach and an improved logistic regression algorithm to predict heart disease, and an accuracy of 89% was obtained. Furthermore, Samuel et al. [29] proposed a novel method for heart disease prediction using a fuzzy analytic hierarchy process (Fuzzy AHP) and an artificial neural network (ANN), and their

technique obtained an accuracy of 91%. Meanwhile, heart disease prediction remains a problem for researchers and scientists because most of the predictive models developed so far have only achieved moderate performance [1].

Autoencoder neural networks have shown remarkable performance in unsupervised feature learning. Several research works have studied the sparse autoencoder (SAE) [20]. Meanwhile, different variants of autoencoders can learn diverse characteristics of the input data. Yang et al. [30] proposed a representation learning approach by serially connecting marginalized autoencoder (MDA) and stacked robust autoencoder with graph regularisation (SRAG) to utilize the learning capabilities of the different autoencoders. The MDA was used to capture domain invariant features, while the SRAG improved the quality of feature representation. Enhanced feature representation was obtained by serially connecting the two autoencoders.

Furthermore, Du et al. [31] developed a sparse autoencoder (SAE) combined with an improved multiple kernel learning strategy to extract high-level discriminative features. The SAE learned hidden representation from facial images and classified the data efficiently. The approach achieved comparable performance with existing facial expression recognition methods. Tai et al. [32] stacked several denoising sparse autoencoders for high-resolution range profile recognition. The training process consists of layer-by-layer pre-training. The training strategy ensured the model converged faster and obtained excellent performance.

In another study, Xiong and Lu [33] proposed a multistage approach to predict Parkinson's disease. The study combined the adaptive grey wolf optimization algorithm and sparse autoencoder. The former was used to identify the predictor candidate subset, and the latter to extract the latent representation of the candidate features for efficient supervised classification. The approach outperformed the benchmark models and showed the importance of representation learning in complex disease diagnosis. Furthermore, Mienye et al. [34] proposed a model to predict heart disease by combining SAE and ANN. The SAE was enhanced to obtain a good feature representation, and the ANN was employed to classify the learned features. The approach achieved a good prediction performance.

Furthermore, Ebiaredoh-Mienye et al. [35] developed an approach to predict diseases using sparse autoencoder and softmax classifier. Unlike other sparse autoencoders that achieve sparsity by penalizing the neural network weights, their technique penalizes the hidden layer activations. When tested on three disease datasets, the method showed improvement over other benchmarked algorithms. Meanwhile, our research aims to obtain an efficient feature representation of the heart disease data by optimizing the parameters of the stacked sparse autoencoder using the particle swarm optimization technique.

3. Materials and Methods

In this section, a description of the heart disease datasets employed in this research is presented. Additionally, this section provides a detailed background of the autoencoder neural network. Lastly, Section 3.3 presents the proposed heart disease prediction methodology, including how the PSO is integrated into the autoencoder architecture with a step-by-step description of the method applied.

3.1. Datasets

This research uses two publicly available heart study datasets, including the Cleveland heart disease dataset [36] and the Framingham heart dataset [37]. The Cleveland dataset consists of 303 samples and 14 attributes, and there are 165 sick patients and 138 healthy patients in the dataset. Meanwhile, the Framingham dataset contains 4238 samples and 16 features, with 3594 healthy patients and 644 sick patients. The attributes of both datasets include risk factors from physical examination of the patients, behavioral risk factors, and medical risk factors. Tables 1 and 2 show the features of the Cleveland and Framingham datasets, respectively.

S/N	Feature	Description
1	Age	The individual's age, in years
2	Sex	The sex of the patient $(1 = male, 0 = female)$
3	Chest pain type	The type of chest pain experienced by the patient $(1 = typical angina, 2 = atypical angina, 3 = non-anginal pain, 4 = asymptomatic)$
4	Resting Blood Pressure	Resting blood pressure value of the patient in mmHg
5	Serum Cholesterol	Serum cholesterol in mg/dL
6	Fasting Blood Sugar	Fasting blood sugar > 120 mg/dL ($1 = true, 0 = false$)
7	Resting ECG	ECG result (0 = normal, 1 = ST-T abnormality, 2 = left ventricular hypertrophy)
8	MaxHeart rate	Maximum heart rate achieved by the patient
9	Exercise induced angina	Exercise induced angina $(1 = true, 0 = false)$
10	ST depression	ST depression induced by exercise relative to rest
11	Slope	The slope of the peak exercise ST segment (1 = upsloping, 2 = flat, 3 = downsloping)
12	Number of vessels	Number of major vessels (0–3) colored by fluoroscopy
13	Thalassemia	Type of thalassemia disorder (3 = normal, 6 = fixed disorder, 7 = reversible disorder)
14	Target variable	The status of heart disease diagnosis $(0 = absence, 1, 2, 3, 4 = present)$

Table 1. Features in the Cleveland datasets.

Table 2. Features in the Framingham dataset.

S/N	Feature	Description
1	Sex	Patient's gender
2	Age	Patient's age
3	Education	Educational level (1 = high school, 2 = GED certificate, 3 = vocational training,
U	Education	4 = college degree)
4	currentSmoker	Whether the individual smokes or not
5	cigsPerDay	The average number of cigarettes the patient smokes per day
6	BPMeds	Whether the individual is on BP medication or not
7	prevalentStroke	Whether the individual previously had a stroke or not
8	PrevalentHyp	Whether the individual is hypertensive
9	Diabetes	Whether the individual is diabetic
10	totChol	Total cholesterol level
11	sysBP	Systolic blood pressure
12	diaBP	Diastolic blood pressure
13	BMI	Body mass index
14	heartrate	The patient's heart rate
15	Glucose	Glucose level
16	Target variable (TenYearCHD)	Whether or not the patient has a ten-year risk of coronary heart disease

The datasets are split into 70% for training and 30% for testing and validation. Both datasets contain missing values, and we utilize mean imputation to handle the missing data. Imputation is a method used to replace missing data with a substitute value based on other available details in order to preserve most of the information in the dataset. Data imputation also ensures proper data are used for machine learning since missing data can be problematic for many machine learning algorithms [38]. Furthermore, we utilize MinMaxScaler to transform the features into the range [0, 1] because the proposed autoencoder involves using a sigmoid activation function that outputs values between 0 and 1.

3.2. Autoencoder

Autoencoders are unsupervised artificial neural networks used for representation learning. The autoencoder architecture is designed to impose a bottleneck in the network that forces a compressed knowledge representation of the input. Therefore, the correlation between the input features is learned and reconstructed. Autoencoders have an encoder-decoder structure [39]. The encoder maps the original input x to the hidden layer, which is considered a latent space representation. The decoder then reconstructs this

latent representation into \hat{x} [20]. The encoding and decoding processes are defined in Equations (1) and (2), respectively.

$$h = \sigma(Wx + b), \tag{1}$$

$$\hat{x} = \sigma(W'h + b'), \tag{2}$$

where $x = (x_1, x_2, ..., x_n)$ represents the input data vector, $h = (h_1, h_2, ..., h_n)$ denotes the low-dimensional vector obtained from the hidden layer, and $\hat{x} = (\hat{x}_1, \hat{x}_2, ..., \hat{x}_n)$ is the reconstructed input. W and W' are weight matrices, b and b' are bias vectors, and σ represents the sigmoid activation function, i.e., $\sigma = \frac{1}{1+e^{-x}}$. We employ the mean squared error function as the reconstruction error function between h and \hat{x} :

$$E = \frac{1}{N} \sum_{i=1}^{N} ||\hat{x}_i - x_i||^2, \qquad (3)$$

Overfitting is a common challenge that occurs when training autoencoder networks. An efficient way to solve this problem is by applying a weight penalty to the cost function [34]:

$$E = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{2} ||\hat{x}_i - x_i||^2 + \frac{\lambda}{2} (||W||^2 + ||W'||^2), \tag{4}$$

where λ is the weight attenuation coefficient. Additionally, a sparse penalty term is introduced in the autoencoder hidden layer to achieve better feature learning under sparse constraints and avoid a situation where the autoencoder copies the input data to the output [40]. Assuming \hat{p}_j denotes the average activation of the hidden layer neurons, it is defined as $\hat{p}_j = \frac{1}{N} \sum_{i=1}^{N} h_j(x_i)$, and ρ is the sparsity proportion, often a small positive value near 0. To achieve sparsity, we limit $\hat{\rho}_j = \rho$ [41], and the Kullback-Leibler (*KL*) divergence is introduced to the loss function as a regularization term:

$$KL(\hat{\rho}||\rho) = \sum_{j=1}^{K} \rho \log\left(\frac{\rho}{\hat{\rho}_j}\right) + (1-\rho) \log\left(\frac{1-\rho}{1-\hat{\rho}_j}\right),$$
(5)

where *K* is the number of hidden neurons. Hence, the loss function of the sparse autoencoder now contains three parts: the mean squared error, the weight attenuation, and the sparsity regularization parts:

$$E = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{2} ||\hat{x}_i - x_i||^2 + \frac{\lambda}{2} (||W||^2 + ||W'||^2) + \beta K L(\hat{\rho}||\rho),$$
(6)

where β is the sparsity regularisation parameter. Furthermore, we stack multiple sparse autoencoders to achieve enhanced feature learning. The architecture entails connecting the encoding layer to the input layer of the next sparse autoencoder, thereby ensuring the network achieves better representation learning.

3.3. Proposed Methodology

In this study, a stacked sparse autoencoder network is proposed. In the SSAE network, the hidden layer of the preceding sparse autoencoder serves as input to the next sparse autoencoder. Additionally, it is essential to note that after training the various sparse autoencoders, their decoder layers are deleted since the learned features reside in the hidden layer. The final hidden layer is then connected to the softmax classifier, which performs the classification. Therefore, the proposed SSAE network consists of the trained sparse autoencoders and softmax classifier. Backpropagation is applied to finetune the parameters of the entire network with the training instances and their labels. The finetuning stage considers the various layers of the network as one model. Assuming $\{y^1, y^2, \ldots, y^m\}$

represents the target variables of the training data, the cost function of the entire network is defined as:

$$E = -\frac{1}{m} \left[\sum_{i=1}^{m} \sum_{j=1}^{N} 1\left\{ y^{i} = j \right\} \log \frac{e^{\theta_{j}^{i} x^{i}}}{\sum_{l=1}^{N} e^{\theta_{l}^{T} x^{i}}} \right],$$
(7)

where $1\{\cdot\}$ represents the indicator function, i.e., $1\{y^i = j\} = 1$ *if* y = 1, and $1\{y^i = j\} = 0$ *if* $y \neq j$, N denotes the number classes, and θ_i denotes the weight matrix linking the *ith* output unit.

Furthermore, we utilize the PSO to optimize the SSAE's parameters, i.e., optimal weights and bias values. The choice of weights and bias is crucial in training robust neural networks. Usually, the determination of optimal network configuration is achieved by trial and error by searching a space of potential hyperparameter combinations [42]. The conventional manual parameter settings and grid search method requires domain knowledge and is error-prone, consumes time, and is computationally expensive [43]. Meanwhile, genetic algorithm (GA) have been used to optimize some neural networks, including autoencoders [44,45]. However, PSO is a less complex algorithm compared to GA, and it converges faster. Hence, in this research, we employ the PSO to optimize the SSAE.

The PSO is a heuristic search method inspired by biological populations' swarming or cooperative behavior [46]. It is an intelligent evolutionary optimization method that iteratively optimizes a problem to enhance a candidate solution regarding a specified quality measure. Every particle is considered a potential solution to the optimization problem and consists of speed, position, and a fitness value computed by the optimization function [47]. Therefore, the introduction of PSO in the SSAE network improves the convergence speed and ensures the autoencoder learns the most representative features.

In the PSO algorithm, there are *m* particles in the swarm. Every particle $p_i(t)$ is randomly initialized to separate positions $x_i(t)$ in the search space, with speed $v_i(t)$. After that, the SSAE would be configured for training by applying the value of $x_i(t)$. Then, the final SAE's reconstruction error (6) is utilized as the PSO's fitness value. The PSO aims to search the space for the minimum fitness value. Assuming $l_i(t)$ represents the fitness value. At every iteration, $l_i(t)$ is computed using Equation (6), and if it is the minimum fitness value that exists in the swarm, then the position $x_i(t)$ is set as the best position *g*. Additionally, if $l_i(t)$ is the minimum fitness value estimated by particle $p_i(t)$, then $x_i(t)$ is used as the best position $b_i(t)$ of the particle. After that, the speed of all the particles is updated based on their location, current speed, and global swarm information. The update of the speed and position of the particles is achieved using Equations (8) and (9), respectively [48]:

$$v(t+1) = \omega v(t) + c_1 r_1(b_i - x(t)) + c_2 r_2(g - x(t)),$$
(8)

$$x(t+1) = x(t) + v(t+1),$$
(9)

where v(t + 1) represent the speed of the particle at time t + 1, ω represents the inertia weight, v(t) is the current speed, c_1 and c_2 are acceleration factors, and their values are 1.5 and 2, respectively, r_1 and r_2 are random numbers between 0 and 1. After the given number of iterations, g contains the best solution, which is utilized to initialize the parameters of the SSAE. Lastly, the algorithm ends the optimization when the evaluation function converges, and then the optimal solution is given as the initial weights and bias; else, the iterations continue. Algorithm 1 summarises the steps followed to achieve the optimization of the SSAE by the PSO.

Algorith	m 1 . Proposed methodology.
Set	the parameters of the PSO, i.e., c_1 , c_2 , ω , the maximum number of iterations T_{max} , and
the	number of particles <i>m</i>
1.	for each particle $p_i(t)$
2.	perform SSAE training using the current particle
3.	compute <i>E</i> using (6) and set fitness value $l_i(t) = E$
4.	update the best position and speed of each particle based on the fitness function, i.e., using (8) and (9)
5.	if the evaluation function converges, then return the optimal solution g as the SSAE weights and bias
6.	else return to step 1 and continue the iteration

Meanwhile, when training deep neural networks such as the SSAE proposed in this paper, internal covariate shift slows down the training process and affects the model's performance. Batch normalization was proposed in 2015 by Ioffe and Szegedy [49] and has been used in recent research works [50,51] to make neural networks faster and more stable via normalizing the layers' inputs through re-scaling and re-centering. Internal covariate shift is the change in the distribution of the neural network activations resulting from the parameter updates during training. It occurs when there is a change in the neural network's input distribution. Usually, when there is a change in input distribution, the hidden layers tend to adapt to this new distribution, which slows down the neural network training. When the training is slowed down, it takes longer for the network to converge to a global minimum. Therefore, we apply batch normalization to improve the SSAE's speed and stability by normalizing layer inputs to prevent this problem. Batch normalization also acts as a regularizer and reduces the need to apply the dropout regularization technique. Similar to the dropout technique, research has shown that batch normalization also prevents overfitting in deep neural networks [52]. Lastly, Figure 1 shows a flowchart of the proposed approach to visually demonstrate all the steps taken to arrive at the PSO optimized SSAE model.



Figure 1. Flowchart of the proposed approach.

In implementing the proposed heart disease prediction method shown in Figure 1, the data are first preprocessed as discussed in Section 3.1. The preprocessed data are split into training and testing sets. The proposed model is pre-trained and finetuned to obtain optimal performance. The pre-training is employed for the SSAE training, whereas the backpropagation is utilized to finetune the model. As shown in Figure 1, several steps are taken to arrive at the final optimal SSAE model.

Step 1: The encoder maps the input data into a low dimensional space, while the decoder reconstructs the data. In order to obtain optimal reconstruction, we employed backpropagation to minimize the reconstruction error. During this step, the bias and weight matrices of the encoder and decoder are obtained using backpropagation.

Step 2: In this step, the sparse autoencoders are stacked to form the SSAE. It is achieved by stacking new hidden and output layers into the first autoencoder, where the hidden layer of the first autoencoder serves as input to the new autoencoder. In this study, only three autoencoders are stacked in order not to make the model too complex.

Step 3: The final hidden layer is then connected to the softmax classifier to complete the SSAE network. Backpropagation is applied to finetune the parameters of the entire network, including the weights and biases.

Step 4: The fourth step involves using PSO to search for the optimal initial weights and bias values of the SSAE. In the PSO implementation, we initialized parameters such as the number of particles, the maximum number of iterations, the acceleration factors, etc., as stated in Algorithm 1. During the backpropagation finetuning in step 3, the PSO is employed to find the optimal SSAE parameters, which leads to enhanced performance.

4. Results and Discussion

This section presents and discusses the results obtained from the experiments. In the experiment, accuracy, precision, sensitivity, and F-measure were used as the performance evaluation metrics, and their formulas are as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN},$$
(10)

$$Precision = \frac{TP}{TP + FP},$$
(11)

$$Sensitivity = \frac{TP}{TP + FN},$$
 (12)

$$F measure = \frac{2 \times precision \times recall}{precision + recall},$$
(13)

where *TP* represents the number of true positives and *TN* denotes the number of true negatives. In contrast, *FP* and *FN* represent the number of false-positive and false negatives, respectively. Other performance evaluation metrics used in this work are the receiver operating characteristic (ROC) curve and the area under the ROC curve (AUC). All the simulations were carried out using the Python programming language and the scikit-learn machine learning library. The experimental results are shown in Tables 3 and 4, where the proposed method is compared with some well-known algorithms. The algorithms include KNN [53], LR [54], LDA [55], SVM [56], decision tree [57], XGBoost [58], random forest [59], softmax [60], and AdaBoost [61]. It is necessary to note that since our problem is that of binary classification, hence, the softmax classifier mentioned above has K = 2 classes. This softmax algorithm is a type of logistic regression that is optimized by SGD. However, it differs from the logistic regression mentioned above as the latter uses the limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) optimization algorithm. Furthermore, we employed the well-known GridSearch tuning technique for hyperparameter tuning of the baseline models to obtain the best performance.

Table 3 shows the performance of the various algorithms when trained with the Framingham dataset. The proposed approach achieved an accuracy of 0.973, a precision of 0.948, a sensitivity of 1.000, and an F-measure of 0.973. From Table 4, the proposed SSAE-PSO method obtained an accuracy of 0.961, precision of 0.930, a sensitivity of 0.988, and an F-measure of 0.958. In both instances, the proposed SSAE-PSO outperformed the other algorithms. However, it is not sufficient to conclude that the proposed method is better using just two datasets. Hence, to further validate the robustness of the proposed

method, we performed experiments using two different disease datasets, i.e., the cervical cancer risk factors [62] and chronic kidney disease [63] datasets, and the results are shown in Tables 5 and 6, respectively.

Algorithm	Accuracy	Precision	Sensitivity	F-Measure
KNN	0.783	0.801	0.789	0.800
LR	0.838	0.839	0.841	0.839
LDA	0.825	0.806	0.828	0.817
SVM	0.805	0.837	0.823	0.830
Decision tree	0.749	0.735	0.745	0.739
Softmax classifier	0.794	0.803	0.786	0.794
XGBoost	0.918	0.938	0.972	0.955
Random forest	0.883	0.911	0.903	0.907
AdaBoost	0.895	0.955	0.889	0.920
Proposed SSAE + PSO	0.973	0.948	1.000	0.973
Proposed SSAE + PSO	0.973	0.948	1.000	0.973

Table 3. Assessment of the performance of the algorithms using the Framingham dataset.

Table 4. Assessment of the performance of the algorithms using the Cleveland dataset.

Algorithm	Accuracy	Precision	Sensitivity	F-Measure
KNN	0.624	0.608	0.594	0.601
LR	0.783	0.790	0.781	0.785
LDA	0.781	0.804	0.792	0.798
SVM	0.796	0.800	0.789	0.794
Decision tree	0.710	0.699	0.708	0.703
Softmax classifier	0.738	0.715	0.700	0.708
XGBoost	0.875	0.864	0.940	0.900
Random forest	0.868	0.914	0.887	0.900
AdaBoost	0.877	0.932	0.865	0.897
Proposed SSAE + PSO	0.961	0.930	0.988	0.958

Table 5. Assessment of the performance of the algorithms using the cervical cancer dataset.

Algorithm	Accuracy	Precision	Sensitivity	F-Measure
KNN	0.956	0.913	0.830	0.870
LR	0.940	0.942	0.978	0.959
LDA	0.942	0.876	0.904	0.890
SVM	0.933	0.918	0.920	0.919
Decision tree	0.892	0.910	0.902	0.906
Softmax classifier	0.938	0.841	0.924	0.881
XGBoost	0.966	0.903	0.917	0.910
Random forest	0.964	0.855	0.912	0.883
AdaBoost	0.955	0.860	0.912	0.885
Proposed SSAE + PSO	0.988	0.984	0.978	0.981

Table 6. Assessment of the performance of the algorithms using the CKD dataset.

Algorithm	Accuracy	Precision	Sensitivity	F-Measure
KNN	0.925	0.910	0.894	0.903
LR	0.927	0.914	0.880	0.897
LDA	0.900	0.896	0.897	0.896
SVM	0.896	0.907	0.900	0.904
Decision tree	0.910	0.921	0.921	0.921
Softmax classifier	0.930	0.924	0.917	0.920
XGBoost	0.940	0.940	0.932	0.936
Random Forest	0.935	0.946	0.930	0.938
AdaBoost	0.947	0.947	0.973	0.960
Proposed SSAE + PSO	0.982	0.974	1.000	0.987

The experimental results in Tables 5 and 6 demonstrate that our approach also obtained superior performance in both instances. Worthy of note is that the proposed method performed better than the ensemble learning algorithms that have obtained state-of-the-art performance in diverse applications. The improved performance in the proposed method can be attributed to the enhanced feature learning in the SSAE, which was made possible by the parameter optimization performed by the PSO.

Furthermore, test sensitivity is an essential metric in medical diagnosis. It measures the proportion of positive or sick patients in the dataset that are correctly predicted [64]. In the Framingham and CKD datasets, the proposed approach obtained sensitivity values of 100%, which implies all the positive samples were correctly predicted in the test set. Additionally, the proposed method achieved a sensitivity of 98.8% and 97.8% for the Cleveland and cervical cancer datasets, respectively. Additionally, the high sensitivity shows the proposed method obtained only a few false negatives. The ROC plots and AUC values of the classifiers in the various test instances are shown below to indicate the superior performance of the proposed approach. We compare the proposed method with the ensemble learning algorithms, which are known to achieve high performance.

Figures 2–5 visibly demonstrate the proposed method's superior performance over some powerful ensemble algorithms in terms of ROC curves and AUC values. This shows the PSO's capability to optimize the SSAE parameters, which resulted in improved feature learning and classification. Furthermore, the proposed method is benchmarked with some recently developed heart disease prediction models that utilized the Cleveland dataset. The methods include an enhanced hybrid random forest [4], an improved logistic regression model with feature selection (FS) [28], an intelligent heart disease prediction approach using a feature selection method and Naïve Bayes model [65], a system based on Fuzzy AHP and ANN [29], an XGBoost with enhanced data resampling technique [2], a method based on mutual information (MI) and deep neural network (DNN) [66], an improved decision tree ensemble [10], a sparse autoencoder combined with ANN [34], a hybrid GA with fuzzy classifier [67], and a fuzzy ensemble system with GA and modified dynamic PSO [68]. The comparison is shown in Table 7. Meanwhile, the Cleveland dataset was chosen because it is a more popular dataset for heart disease prediction and has been widely utilized in numerous studies.

The classification accuracy of 96.1% makes the proposed approach better than some state-of-the-art methods tabulated in Table 7. An essential contribution of this work is that the proposed method learns efficiently using few samples, which is vital because data collection is a complex and expensive task. Hence, a technique that can learn from a few instances is more desirable. Furthermore, the experimental results obtained from this research have shown that the proposed PSO based stacked sparse autoencoder yields efficient representation learning of the input data, which resulted in improved classification performance.



Figure 2. ROC curve of the classifiers using the Framingham dataset.







Figure 4. ROC curve of the classifiers using the cervical cancer dataset.



Figure 5. ROC curve of the classifiers using the CKD dataset.

 Table 7. Performance comparison with other heart disease studies.

Author(s)	Method	Accuracy	Precision	Sensitivity	F-Measure
Mohan et al. [4]	Hybrid random forest	0.884	0.901	0.928	0.900
Haq et al. [28]	Improved Logistic Regression Model + FS	0.890	-	0.770	-
Repaka et al. [65]	Naïve Bayes + FS	0.8977	-	-	-
Samuel et al. [29]	Fuzzy AHP + ANN	0.910	-	-	-
Ali et al. [69]	Optimized SVM	0.922	0.829	1.000	-
Li et al. [5]	FS + SVM	0.923	-	0.98	-

Author(s)	Method	Accuracy	Precision	Sensitivity	F-Measure
Fitriyani et al. [2]	XGBoost + Resampling	0.984	0.985	0.983	0.983
Ali and Bukhari [66]	MI + DNN	0.933	-	0.902	-
Mienye et al. [10]	Ensemble learning	0.930	0.960	0.910	0.930
Mienye et al. [34]	SAE + ANN	0.900	0.890	0.910	0.900
Reddy et al. [67]	Hybrid GA + Fuzzy Classifier	0.900	-	0.910	-
Paul et al. [68]	Ensemble Fuzzy system + GA + PSO	0.923	-	-	-
Our approach	SSAE + PSO	0.961	0.930	0.988	0.958

Table 7. Cont.

Meanwhile, heart disease detection is considered among the most vital studies in clinical data analysis [70], and its prediction is difficult due to the many contributory risk factors. Hence, researchers and scientists have studied different ML methods to accurately predict this disease. This study's optimized feature learning method ensures that the input features' salient relationships are learned to achieve efficient and reliable prediction. Thus, the proposed approach could assist clinicians in the early detection of disease risk when deployed as a clinical decision support system, especially in developing countries with limited healthcare infrastructure. Since there is a high possibility of curing diseases when they are detected early, clinicians can use tools like this for early detection and adequately proffer the needed treatment and recommend lifestyle changes where necessary.

5. Conclusions

Heart disease prediction is a critical challenge in clinical machine learning. This research proposed an effective heart disease prediction method using a stacked sparse autoencoder with a softmax layer. In the proposed approach, the hidden layer of the last sparse autoencoder was connected to the softmax classifier, which made up the SSAE network. A notable contribution in this study is the integration of the PSO to optimize the parameters of the SSAE, which enhanced the feature learning and improved the classification performance. The approach introduced in this study ensures the problem of internal covariate shift associated with multilayer autoencoders such as the SSAE is prevented. When experimented on the Framingham and Cleveland heart datasets, the proposed method achieved an accuracy of 0.973 and 0.961, respectively, which showed significant improvement compared to other ML algorithms and recent studies. In the future, we will study the impact of feature learning on different classification algorithms and focus on stacking different variants of autoencoders and observing the effect on classification performance.

Author Contributions: Conceptualization, I.D.M. and Y.S.; methodology, I.D.M. and Y.S.; software, I.D.M.; validation, I.D.M. and Y.S.; formal analysis, I.D.M. and Y.S.; investigation, I.D.M. and Y.S.; resources, Y.S.; data curation, I.D.M.; writing—original draft preparation, I.D.M.; writing—review and editing, Y.S.; visualization, I.D.M.; supervision, Y.S.; project administration, Y.S.; funding acquisition, Y.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the South African National Research Foundation under Grant 120106 and Grant 132797, and in part by the South African National Research Foundation Incentive under Grant 132159.

Conflicts of Interest: The authors declare no conflict of interest.

References

- 1. Chicco, D.; Jurman, G. Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. *BMC Med. Inform. Decis. Mak.* **2020**, *20*, 16. [CrossRef]
- Fitriyani, N.L.; Syafrudin, M.; Alfian, G.; Rhee, J. HDPM: An Effective Heart Disease Prediction Model for a Clinical Decision Support System. *IEEE Access* 2020, *8*, 133034–133050. [CrossRef]

- Benjamin, E.J.; Muntner, P.; Alonso, A.; Bittencourt, M.S.; Callaway, C.W.; Carson, A.P.; Chamberlain, A.M.; Chang, A.R.; Cheng, S.; Das, S.R.; et al. Heart Disease and Stroke Statistics-2019 Update: A Report From the American Heart Association. *Circulation* 2019, 139, e56–e528. [CrossRef]
- 4. Mohan, S.; Thirumalai, C.; Srivastava, G. Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques. *IEEE Access* 2019, 7, 81542–81554. [CrossRef]
- 5. Li, J.P.; Haq, A.U.; Din, S.U.; Khan, J.; Khan, A.; Saboor, A. Heart Disease Identification Method Using Machine Learning Classification in E-Healthcare. *IEEE Access* 2020, *8*, 107562–107582. [CrossRef]
- Mdhaffar, A.; Bouassida Rodriguez, I.; Charfi, K.; Abid, L.; Freisleben, B. CEP4HFP: Complex Event Processing for Heart Failure Prediction. *IEEE Trans. NanoBioscience* 2017, 16, 708–717. [CrossRef]
- 7. Jin, B.; Che, C.; Liu, Z.; Zhang, S.; Yin, X.; Wei, X. Predicting the Risk of Heart Failure With EHR Sequential Data Modeling. *IEEE Access* 2018, *6*, 9256–9261. [CrossRef]
- Ali, L.; Rahman, A.; Khan, A.; Zhou, M.; Javeed, A.; Khan, J.A. An Automated Diagnostic System for Heart Disease Prediction Based on x² Statistical Model and Optimally Configured Deep Neural Network. *IEEE Access* 2019, 7, 34938–34945. [CrossRef]
- Alaa, A.M.; Bolton, T.; Angelantonio, E.D.; Rudd, J.H.F.; van der Schaar, M. Cardiovascular disease risk prediction using automated machine learning: A prospective study of 423,604 UK Biobank participants. *PLoS ONE* 2019, 14, e0213653. [CrossRef] [PubMed]
- 10. Mienye, I.D.; Sun, Y.; Wang, Z. An improved ensemble learning approach for the prediction of heart disease risk. *Inform. Med. Unlocked* **2020**, *20*, 100402. [CrossRef]
- Buchan, T.A.; Ross, H.J.; McDonald, M.; Billia, F.; Delgado, D.; Duero Posada, J.G.; Luk, A.; Guyatt, G.H.; Alba, A.C. Physician Prediction versus Model Predicted Prognosis in Ambulatory Patients with Heart Failure. *J. Heart Lung Transplant.* 2019, 38, S381. [CrossRef]
- Oh, S.L.; Jahmunah, V.; Ooi, C.P.; Tan, R.-S.; Ciaccio, E.J.; Yamakawa, T.; Tanabe, M.; Kobayashi, M.; Acharya, U.R. Classification of heart sound signals using a novel deep WaveNet model. *Comput. Methods Programs Biomed.* 2020, 105604. [CrossRef] [PubMed]
- Bengio, Y.; Courville, A.; Vincent, P. Representation Learning: A Review and New Perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* 2013, 35, 1798–1828. [CrossRef] [PubMed]
- 14. Kasongo, S.M.; Sun, Y. A Deep Learning Method With Filter Based Feature Engineering for Wireless Intrusion Detection System. *IEEE Access* **2019**, *7*, 38597–38607. [CrossRef]
- 15. Kasongo, S.M.; Sun, Y. Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset. J. Big Data 2020, 7, 105. [CrossRef]
- 16. Reddy, G.T.; Reddy, M.P.K.; Lakshmanna, K.; Kaluri, R.; Rajput, D.S.; Srivastava, G.; Baker, T. Analysis of Dimensionality Reduction Techniques on Big Data. *IEEE Access* 2020, *8*, 54776–54788. [CrossRef]
- 17. Wickramasinghe, C.S.; Marino, D.L.; Manic, M. ResNet Autoencoders for Unsupervised Feature Learning From High-Dimensional Data: Deep Models Resistant to Performance Degradation. *IEEE Access* **2021**, *9*, 40511–40520. [CrossRef]
- 18. Zhang, C.; Cheng, X.; Liu, J.; He, J.; Liu, G. Deep Sparse Autoencoder for Feature Extraction and Diagnosis of Locomotive Adhesion Status. *J. Control. Sci. Eng.* **2018**, *2018*, 8676387. [CrossRef]
- 19. Ng, A. Sparse Autoencoder. 2011. Available online: https://web.stanford.edu/class/cs294a/sparseAutoencoder.pdf (accessed on 6 June 2020).
- 20. Liu, J.; Li, C.; Yang, W. Supervised Learning via Unsupervised Sparse Autoencoder. IEEE Access 2018, 6, 73802–73814. [CrossRef]
- 21. Mienye, I.D.; Ainah, P.K.; Emmanuel, I.D.; Esenogho, E. Sparse noise minimization in image classification using Genetic Algorithm and DenseNet. In Proceedings of the 2021 Conference on Information Communications Technology and Society (ICTAS), Durban, South Africa, 10–11 March 2021; pp. 103–108. [CrossRef]
- 22. Mienye, I.D.; Sun, Y.; Wang, Z. Improved Predictive Sparse Decomposition Method with Densenet for Prediction of Lung Cancer. *Int. J. Comput.* 2020, 533–541. [CrossRef]
- Lin, C.-J.; Jeng, S.-Y. Optimization of Deep Learning Network Parameters Using Uniform Experimental Design for Breast Cancer Histopathological Image Classification. *Diagnostics* 2020, 10, 662. [CrossRef]
- 24. Kaur, S.; Aggarwal, H.; Rani, R. Hyper-parameter optimization of deep learning model for prediction of Parkinson's disease. *Mach. Vis. Appl.* **2020**, *31*, 32. [CrossRef]
- 25. Bickel, S.; Brückner, M.; Scheffer, T. Discriminative Learning Under Covariate Shift. J. Mach. Learn. Res. 2009, 10, 2137–2155.
- Pasha, S.J.; Mohamed, E.S. Novel Feature Reduction (NFR) Model With Machine Learning and Data Mining Algorithms for Effective Disease Risk Prediction. *IEEE Access* 2020, *8*, 184087–184108. [CrossRef]
- Ali, S.A.; Raza, B.; Malik, A.K.; Shahid, A.R.; Faheem, M.; Alquhayz, H.; Kumar, Y.J. An Optimally Configured and Improved Deep Belief Network (OCI-DBN) Approach for Heart Disease Prediction Based on Ruzzo–Tompa and Stacked Genetic Algorithm. *IEEE Access* 2020, *8*, 65947–65958. [CrossRef]
- 28. Haq, A.U.; Li, J.P.; Memon, M.H.; Nazir, S.; Sun, R. A Hybrid Intelligent System Framework for the Prediction of Heart Disease Using Machine Learning Algorithms. *Mob. Inf. Syst.* **2018**, 2018, 3860146. [CrossRef]
- 29. Samuel, O.W.; Asogbon, G.M.; Sangaiah, A.K.; Fang, P.; Li, G. An integrated decision support system based on ANN and Fuzzy_AHP for heart failure risk prediction. *Expert Syst. Appl.* **2017**, *68*, 163–172. [CrossRef]
- 30. Yang, S.; Zhang, Y.; Zhu, Y.; Li, P.; Hu, X. Representation learning via serial autoencoders for domain adaptation. *Neurocomputing* **2019**, *351*, 1–9. [CrossRef]

- 31. Du, L.; Wu, Y.; Hu, H.; Wang, W. Self-adaptive weighted synthesised local directional pattern integrating with sparse autoencoder for expression recognition based on improved multiple kernel learning strategy. *IET Comput. Vis.* **2020**, *14*, 73–83. [CrossRef]
- 32. Tai, G.; Wang, Y.; Li, Y.; Hong, W. Radar HRRP target recognition based on stacked denosing sparse autoencoder. *J. Eng.* **2019**, 2019, 7945–7949. [CrossRef]
- 33. Xiong, Y.; Lu, Y. Deep Feature Extraction From the Vocal Vectors Using Sparse Autoencoders for Parkinson's Classification. *IEEE Access* 2020, *8*, 27821–27830. [CrossRef]
- 34. Mienye, I.D.; Sun, Y.; Wang, Z. Improved sparse autoencoder based artificial neural network approach for prediction of heart disease. *Inform. Med. Unlocked* 2020, *18*, 100307. [CrossRef]
- 35. Ebiaredoh-Mienye, S.A.; Esenogho, E.; Swart, T.G. Integrating Enhanced Sparse Autoencoder-Based Artificial Neural Network Technique and Softmax Regression for Medical Diagnosis. *Electronics* **2020**, *9*, 1963. [CrossRef]
- UCI Machine Learning Repository: Heart Disease Data Set. Available online: http://archive.ics.uci.edu/ml/datasets/Heart+ Disease (accessed on 9 April 2020).
- 37. Framingham Heart Study Dataset. Available online: https://kaggle.com/amanajmera1/framingham-heart-study-dataset (accessed on 24 January 2020).
- 38. Khan, S.I.; Hoque, A.S.M.L. SICE: An improved missing data imputation technique. J. Big Data 2020, 7, 37. [CrossRef]
- 39. Pathirage, C.S.N.; Li, J.; Li, L.; Hao, H.; Liu, W.; Wang, R. Development and application of a deep learning–based sparse autoencoder framework for structural damage identification. *Struct. Health Monit.* **2019**, *18*, 103–122. [CrossRef]
- 40. Yan, B.; Han, G. Effective Feature Extraction via Stacked Sparse Autoencoder to Improve Intrusion Detection System. *IEEE Access* 2018, *6*, 41238–41248. [CrossRef]
- 41. Li, G.; Han, D.; Wang, C.; Hu, W.; Calhoun, V.D.; Wang, Y.-P. Application of deep canonically correlated sparse autoencoder for the classification of schizophrenia. *Comput. Methods Programs Biomed.* **2020**, *183*, 105073. [CrossRef] [PubMed]
- 42. Doaud, M.; Mayo, M. Using Swarm Optimization To Enhance Autoencoders Images. arXiv 2018, arXiv:1807.03346.
- 43. Fernandes Junior, F.E.; Yen, G.G. Particle swarm optimization of deep neural networks architectures for image classification. *Swarm Evol. Comput.* **2019**, *49*, 62–74. [CrossRef]
- 44. Chiroma, H.; Noor, A.S.M.; Abdulkareem, S.; Abubakar, A.I.; Hermawan, A.; Qin, H.; Hamza, M.F.; Herawan, T. Neural Networks Optimization through Genetic Algorithm Searches: A Review. *Appl. Math. Inf. Sci.* **2017**, *11*, 1543–1564. [CrossRef]
- 45. Feng, X.; Zhao, J.; Kita, E. Genetic Algorithm-based Optimization of Deep Neural Network Ensemble. *Rev. Socionetwork Strat* **2021**, *15*, 27–47. [CrossRef]
- Yang, X.-S. Chapter 8-Particle Swarm Optimization. In Nature-Inspired Optimization Algorithms, 2nd ed.; Yang, X.-S., Ed.; Academic Press: Cambridge, MA, USA, 2021; pp. 111–121.
- 47. Kennedy, J. Particle Swarm Optimization. In *Encyclopedia of Machine Learning*; Sammut, C., Webb, G.I., Eds.; Springer: Boston, MA, USA, 2010; pp. 760–766.
- Qolomany, B.; Maabreh, M.; Al-Fuqaha, A.; Gupta, A.; Benhaddou, D. Parameters optimization of deep learning models using Particle swarm optimization. In Proceedings of the 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC), Valencia, Spain, 26–30 June 2017; pp. 1285–1290. [CrossRef]
- 49. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv* 2015, arXiv:1502.03167.
- Hiriyannaiah, S.; Srinivas, A.M.D.; Shetty, G.K.; Siddesh, G.M.; Srinivasa, K.G. Chapter 4-A computationally intelligent agent for detecting fake news using generative adversarial networks. In *Hybrid Computational Intelligence*; Bhattacharyya, S., Snášel, V., Gupta, D., Khanna, A., Eds.; Academic Press: Cambridge, MA, USA, 2020; pp. 69–96.
- Theodoridis, S. Chapter 18-Neural Networks and Deep Learning. In *Machine Learning*, 2nd ed.; Theodoridis, S., Ed.; Academic Press: Cambridge, MA, USA, 2020; pp. 901–1038. [CrossRef]
- 52. Garbin, C.; Zhu, X.; Marques, O. Dropout vs. batch normalization: An empirical study of their impact to deep learning. *Multimed Tools Appl.* **2020**, *79*, 12777–12815. [CrossRef]
- 53. Altman, N.S. An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression. Am. Stat. 1992, 46, 175–185. [CrossRef]
- 54. Cramer, J.S. The Origins of Logistic Regression; Social Science Research Network: Rochester, NY, USA, 2002. [CrossRef]
- Hastie, T.; Tibshirani, R.; Friedman, J. Linear Methods for Classification. In *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Hastie, T., Tibshirani, R., Friedman, J., Eds.; Springer: New York, NY, USA, 2009; pp. 101–137. [CrossRef]
- Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* 1995, 20, 273–297. [CrossRef]
 Krzywinski, M.; Altman, N. Classification and regression trees. *Nat. Methods* 2017, 14, 757–758. [CrossRef]
- Breiman, L. Arcing the Edge; Technical Report 486; Statistics Department, University of California: Los Angeles, CA, USA, 1997.
- 59. Breiman, L. Random Forests. Mach. Learn. 2001, 45, 5–32. [CrossRef]
- 60. Bridle, J.S. Probabilistic Interpretation of Feedforward Classification Network Outputs, with Relationships to Statistical Pattern Recognition. In *Neurocomputing*; Springer: Berlin/Heidelberg, Germany, 1990; pp. 227–236. [CrossRef]
- 61. Schapire, R.E. A brief introduction to boosting. In Proceedings of the 16th Internation Joint Conference on Artificial Intelligence, IJCAI, Stockholm, Sweden, 31 July–6 August 1999; Volume 2, pp. 1401–1406.
- 62. UCI Machine Learning Repository: Cervical Cancer (Risk Factors) Data Set. Available online: https://archive.ics.uci.edu/ml/datasets/Cervical+cancer+%28Risk+Factors%29 (accessed on 15 April 2021).

- 63. UCI Machine Learning Repository: Chronic_Kidney_Disease Data Set. Available online: https://archive.ics.uci.edu/ml/ datasets/Chronic_Kidney_Disease (accessed on 20 July 2021).
- 64. Mienye, I.D.; Sun, Y. Performance analysis of cost-sensitive learning methods with application to imbalanced medical data. *Inform. Med. Unlocked* **2021**, *25*, 100690. [CrossRef]
- Repaka, A.N.; Ravikanti, S.D.; Franklin, R.G. Design And Implementing Heart Disease Prediction Using Naives Bayesian. In Proceedings of the 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 23–25 April 2019; pp. 292–297. [CrossRef]
- 66. Ali, L.; Bukhari, S.A.C. An Approach Based on Mutually Informed Neural Networks to Optimize the Generalization Capabilities of Decision Support Systems Developed for Heart Failure Prediction. *IRBM* **2020**. [CrossRef]
- 67. Reddy, G.T.; Reddy, M.P.K.; Lakshmanna, K.; Rajput, D.S.; Kaluri, R.; Srivastava, G. Hybrid genetic algorithm and a fuzzy logic classifier for heart disease diagnosis. *Evol. Intel.* 2020, *13*, 185–196. [CrossRef]
- Paul, A.K.; Shill, P.C.; Rabin, M.R.I.; Murase, K. Adaptive weighted fuzzy rule-based system for the risk level assessment of heart disease. *Appl. Intell.* 2018, 48, 1739–1756. [CrossRef]
- Ali, L.; Niamat, A.; Khan, J.A.; Golilarz, N.A.; Xingzhong, X.; Noor, A.; Nour, R.; Bukhari, S.A.C. An Optimized Stacked Support Vector Machines Based Expert System for the Effective Prediction of Heart Failure. *IEEE Access* 2019, 7, 54007–54014. [CrossRef]
- Kim, J.O.R.; Jeong, Y.-S.; Kim, J.H.; Lee, J.-W.; Park, D.; Kim, H.-S. Machine Learning-Based Cardiovascular Disease Prediction Model: A Cohort Study on the Korean National Health Insurance Service Health Screening Database. *Diagnostics* 2021, 11, 943. [CrossRef] [PubMed]